REPORT DOCUMENTATION PAGE

Form Approved
OMB NO. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	July 1, 1999	Final 2/1/95-1/31/99

4. TITLE AND SUBTITLE
Parallel Adaptive Techniques for Transient Partial Differential Equations

5. FUNDING NUMBERS
DAAH04-95-1-0091

6. AUTHOR(S)
Joseph E. Flaherty & Mark S. Shephard

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Rensselaer Polytechnic Institute
110-8th Street Troy, New York 12180-3590

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U. S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
ARO 33095.20-MA

11. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.

12 a. DISTRIBUTION / AVAILABILITY STATEMENT	12 b. DISTRIBUTION CODE
Approved for public release; distribution unlimited.	

13. ABSTRACT (Maximum 200 words)

Our research aims to develop a framework for adaptive and parallel computation on geometrically complex regions. In particular, we investigated dynamic load balancing, transient solution techniques, and error estimation procedures for adaptive computation. Load balancing includes geometrically- and topologically-based procedures that are suitable for heterogeneous computation involving *p*- and *hp*-refinement, time dependence including local time stepping and method orders, diverse computing systems ({*e.g.*, clusters of workstations), and hierarchical networks ({*e.g.*, networks of SMPs). Appropriate time integration techniques include explicit methods and implicit one- and multi-step methods. The explicit methods are useful for problems having very rapid dynamics. Implicit multistep methods are generally more efficient than one-step methods; however, this need not be the case when local time steps and method orders are used. {*A posteriori* error estimation focus on procedures for transient problems with emphasis on singularly-perturbed parabolic and hyperbolic problems, high-order methods involving *p*- and *hp*-refinement, and the coordination of spatial and temporal errors.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Adaptive Methods Parallel Computation Finite Element Methods	10
	16. PRICE CODE

17. SECURITY CLASSIFICATION OR REPORT	18. SECURITY CLASSIFICATION ON THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Standard Form 298 (R...

Final Progress Report

Parallel Adaptive Techniques for Transient Partial Differential Equations

Contract Number DAAH04-95-1-0091

1 February 1995 - 31 January 1999

Joseph E. Flaherty
and
Mark S. Shephard

Scientific Computation Research Center
Rensselaer Polytechnic Institute
Troy, New York 12180, USA

1 Statement of the Problem

Our research aims to develop a framework for adaptive and parallel computation [30, 31] on geometrically complex regions. In particular, we investigated dynamic load balancing, transient solution techniques, and error estimation procedures for adaptive computation. Load balancing includes geometrically- and topologically-based procedures that are suitable for heterogeneous computation involving p- and hp-refinement, time dependence including local time stepping and method orders, diverse computing systems (*e.g.*, clusters of workstations), and hierarchical networks (*e.g.*, networks of SMPs). Appropriate time integration techniques include explicit methods and implicit one- and multi-step methods. The explicit methods are useful for problems having very rapid dynamics. Implicit multistep methods are generally more efficient than one-step methods; however, this need not be the case when local time steps and method orders are used. *A posteriori* error estimation focus on procedures for transient problems with emphasis on singularly-perturbed parabolic and hyperbolic problems, high-order methods involving p- and hp-refinement, and the coordination of spatial and temporal errors.

2 Summary of Important Results

2.1 Geometry-Based Object-Oriented Simulation Framework

To date, consideration of object-oriented programming in simulation software has focused on flexible structures with code reuse, application of symbolic computing, operating in parallel, linking with design processes, and supporting interacting multiphysics simulations. Building on these efforts and the needs of adaptive simulation technologies, we have constructed a geometry-based simulation frameworks that supports parallel adaptive simulation capabilities. This system, referred to as Trellis is based on [6, 5]:

- A set of geometry-based structures which can support; (i) the direct linkage with company CAD information, (ii) all forms of adaptivity without introducing geometric approximation errors, and (iii) the high level integration of multiscale, multi-physics analysis methodologies.

- A careful decomposition of the geometry, physics, mathematical model, discretization and numerical methods into interacting classes. These structures support a variety of equation discretization methods. Both finite element [6, 5] and partition of unity methods have been implemented [25, 24].

- Adaptive control of each step of the simulation process from the selection of the mathematical model, through the model and domain discretization, to the selection of application of the numerical methods to solving the discrete system.

Conceptually Trellis is built on the view of an analysis as a transformation between three levels of description. The highest level description is that of the physical problem which is posed in terms of physical objects interacting with their environment. Since the goal of the analysis is to obtain reliable estimates of the response of the system the second level is a mathematical problem description that introduces some level of idealization, which also needs to be controlled to yield the desired accuracy. The third level is the numerical discretization constructed from a mathematical problem that involves another set of idealizations which also need to be controlled.

The structures used to support the problem definition, the discretizations of the model and their interactions are central to Trellis. The two structures of the geometric model and attributes are used to house the problem definition. The analysis discretizations are housed in the mesh structure. The final structure is the field structure which houses numerical solution results.

The geometric model representation is a non-manifold boundary representation based. The representation used for a mesh is similar to that used for a geometric model: a hierarchy of regions, faces, edges and vertices. In addition, each mesh entity maintains a relation, called classification, to the model entity that it was created to partially represent. Understanding how the mesh relates to the geometric model is critical for both mesh adaptivity and understanding how the solution relates back to the original problem description. The topological representation can also be used to great advantage in performing adaptive p-version analyses as polynomial orders can be directly assigned to the various entities.

A problem with many "classic" numerical analysis codes is that the solution of an analysis is given in terms of the values at a set of discrete points. Trellis eliminates this problem by introducing a construct known as a field which describes the variation of a tensor over one or more entities in a geometric model. The spatial variation of the field is defined in terms of interpolations defined over a discrete representation of the geometric model entities, which can be a mesh.

The Trellis analysis process is a series of transformations of the problem from the original mathematical problem description through to sets of algebraic equations approximately representing the problem. The mathematical problem description level is described by a ContinuousSystem class, which contains the geometric model and the attributes which apply to that model, specified by a particular case node in the attribute graph. An instance of a ContinuousSystem is then transformed to an instance of the class DiscreteSystem which represents the discretized version of the model and attributes and the weak form of the partial differential equation (PDE). The particular analysis class that is used depends on the selected weak form of the PDE to be solved.

The DiscreteSystem class represents the problem in terms of contributions from a set of objects that live on the discrete representation of the model. These objects are called SystemContributors. There are three types of SystemContributors: StiffnessContributors contribute coupling terms between degrees of freedom of the system, ForceContributors contribute terms to the right hand side vector, and Constraints set specific values or constraints to given degrees of freedom. These objects are created by the Analysis object and correspond to an interpretation of attributes consistent with the weak form that the Analysis implements.

The Analysis class creates all of the SystemContributors and adds them to an instance of a DiscreteSystem. The DiscreteSystem is transformed into an AlgebraicSystem, an Assembler object. Multiple linear solvers can be used to solve the AlgebraicSystem. The most extensive capability included is the Portable, Extensible Toolkit for Scientific Computation (PETSc) from Argonne National Laboratory. These procedures have the dual advantage of working effectively in an object-oriented analysis framework and providing an efficient set of linear algebra routines.

Trellis is not complete, but is being used to address complex problems involving compressible flows [20, 18, 19, 17, 16] and other problems. Steady and transient compressible flow problems may be solved by a Galerkin space-time finite element formulation [30] with a least squares stabilization. Compressible flow problems with more transient effects are solved by a discontinuous Galerkin method with explicit time integration and local time stepping [18]. This method is proving to be very efficient since small time steps are restricted to regions containing shocks, expansions, and other nonuniformities. Three-dimensional acoustics problems [28, 34] are treated by high-order methods with h- and p-refinement. The same base software, including the adaptive and parallel procedures, can handle these diverse applications.

Example 1. We illustrate some capabilities of the Trellis framework by using the explicit discontinuous Galerkin software [18, 20] to address the three-dimensional unsteady flow of a compressible gas in a circular cylinder that contains a cylindrical vent. This problem was motivated by shock tube studies as part of an investigation of perforated muzzle brakes for large-calibre cannons. Our focus is on the quasi-stationary flow that exists behind the contact surface (diaphragm of the shock tube) for a short time. Thus, we initiate the problem with a Mach 1.23 flow of helium in the main tube and a quiet flow in the vent. A hypothetical diaphragm between the vent and the main tube is ruptured at time zero to direct the flow into the vent. Using symmetry to divide the problem in half, we calculate solutions on 16 processors of an IBM SP2 at the Rensselaer Polytechnic Institute using local h-refinement with an initial mesh of 80,659 tetrahedral elements. Mesh partitioning was done by a parallel octree traversal at each adaptive h-refinement step.

In Figure 1, we show projections of the Mach number and velocity vectors (left) and of the the mesh and partitioning used onto the symmetry plane and cylindrical surfaces near the vent at three (dimensionless) times.

The flow accelerates as it enters the vent. A strong shock forms near the downwind vent-shock tube interface and a portion of the flow in the vent accelerates to supersonic speeds. The reflection of the flow from the downwind vent face produces a component of the flow at the vent exit in a direction opposite to the principal flow direction. In a cannon, this reduces recoil. Flow features compare well with experiments and earlier computational results. The mesh is concentrated in the shock and expansion regions and remains so as these features evolve. Likewise, an initially (quasi) uniform partitioning is adjusted to contain the smaller elements formed in the shock and expansion regions. Variable time steps that are smaller on the smaller elements of the mesh are not visible in the figure. This three-dimensional unsteady problem would be difficult to solve without such adaptivity.

2.2 Dynamic Load Balancing

We developed three dynamic load balancing schemes: iterative tree balancing [20, 31, 30], parallel sort inertial bisection (PSIRB) [31, 30], and octree partitioning (OCTPART) [20, 18, 21]. The first is iterative (incremental) and the latter two are direct (global). All execute in parallel on a spatially-distributed mesh as part of the adaptive system. Performance of all load balancing procedures can be enhanced by partition boundary smoothing, predictive load balancing, and weighting. Weighting accounts for the complexities of adaptive p-refinement, local time stepping, and heterogeneous computing environments. Partition boundary smoothing [20] eliminates elements that protrude into another partition; thus, unnecessarily increasing the number of element faces on partition boundaries. These can be reduced by a single boundary traversal, which is both inexpensive and effective. Predictive load balancing uses the enrichment schedule to anticipate and correct an imbalance before element migration proceeds [20, 18, 19, 21]. It provides a better balance during migration, saves time by, typically, working on a coarser mesh prior to refinement, and often avoids the need for rebalancing during the subsequent computational step. The procedure has been able to reduce rebalancing times by as much as 35% and total computational time by as much as 20% [20].

2.3 Parallel Data Management

We are developing a system called the Rensselaer Partition Model (RPM) to manage heterogeneous computation [33]. The system is hierarchical with a *Partition Model* describing how the domain is divided for computation. Each mesh entity is classified on a single partition model entity. The partition model is a topological construct similar in concept to the boundary representation used to represent the domain geometry and mesh. The *Process Model* is a disconnected set of nodes with each node representing a single process or thread of execution. Each process model entity can control one or more partition model entities. The *Machine Model* describes the computational environment where the process is running. The model is a graph that describes the various hierarchies of the computational environment. Each terminal node of the graph represents a single CPU that has certain properties such as available memory and computational power. Higher nodes in the graph represent the grouping of processors into SMP nodes or computer clusters. Each processor can communicate with any other processor; however, preferential communication paths are represented in the graph.

2.4 *A Posteriori* Error Estimation

Our concentration has been on developing *a posteriori* error estimates for singularly-perturbed reaction- and convection-diffusion problems. We have developed very simple estimates that use an odd-even dichotomy principle of Babuška. The errors of odd-order finite element approximations arise near element boundaries and may be computed from jumps in solution gradients at element vertices. Conversely, the errors of even-order approximations arise in element interiors and may be computed by solving element-level finite element problems using "bubble functions" that vanish on element boundaries [3]. Error estimates computed in this manner are asymptotically correct on rectangular elements. Evidence suggests that they are also correct for triangular and quadrilateral [2] elements. Error estimates developed for product spaces [3] have been extended to modified hierarchical spaces [2].

We have showed that the spatial discretization errors of hyperbolic problems could be estimated by using Radau polynomials. This was based on earlier work of Adjerid *et al.* [1] who showed that the Radau points are superconvergence points for convection-diffusion problems in the limit of vanishing diffusion. These results provide simple error estimation procedures for hyperbolic problems.

2.5 Solution Procedures

We have continued to investigate preconditioning techniques for large, sparse, linear systems. In doing this, our efforts have focused on the major area of applying the AMLI preconditioner in parallel. We have generated and tested a preliminary version of a parallel, p-level (polynomial-level) preconditioner and have demonstrated its utility in parallel [34]. We are working on performance and scalability improvements to this algorithm.

3 List of Publications

[1] S. Adjerid, M. Aiffa, and J.E. Flaherty. High-order finite element methods for singularly-perturbed elliptic and parabolic problems. *SIAM Journal on Applied Mathematics*, 55:520–543, 1995.

[2] S. Adjerid, B. Belguendouz, and J.E. Flaherty. A posteriori finite element error estimation for diffusion problems. *SIAM Journal on Scientific Computing*, 1998. to appear.

[3] S. Adjerid, J.E. Flaherty, and I. Babuška. A posteriori error estimation for the finite element method-of-lines solution of parabolic problems. *Mathematical Models and Methods in Applied Science*, 9:261–286, 1999.

[4] I. Babuška, J.E. Flaherty, W.D. Henshaw, J.E. Hopcroft, J.E. Oliger, and T. Tezduyar, editors. *Modeling, Mesh Generation, and Adaptive Numerical Methods for Partial Differential Equations*, volume 75 of *The IMA Volumes in Mathematics and its Applications*. Springer-Verlag, New York, 1995. 450 pages.

[5] M.W. Beall. *An Object-Oriented Framework for the Reliable Automated Solution of Problems in Mathematical Physics*. PhD thesis, Rensselaer Polytechnic Institute, Troy, 1999. Scientific Computation Research Center, 6-1999.

[6] M.W. Beall and M.S. Shephard. An object-oriented framework for reliable numerical simulations. *Engineering with Computers*, 15:61–72, 1999.

[7] M. Benantar, J.E. Flaherty, U. Dogrusoz, and M.S. Krishnamoorthy. Triangle graphs. *Applied Numerical Mathematics*, 17:85–96, 1995.

[8] C.L. Bottasso, O. Klaas, and M.S. Shephard. Data structures and mesh modifications for unstructured multigrid adaptive techniques. *Engineering with Computers*, 14:235–247, 1998.

[9] H.L. de Cougny. *Parallel Unstructured Distributed Three-Dimensional Mesh Generation*. PhD thesis, Rensselaer Polytechnic Institute, Troy, 1998. Scientific Computation Research Center, 6-1998.

[10] H.L. de Cougny. Refinement and coarsening of surface meshes. *Engineering with Computers*, 14:214–222, 1998.

[11] H.L. de Cougny and M.S. Shephard. Surface meshing using vertex insertion. In *Proceedings 5th International Meshing Roundtable '96*, volume SAND96-230, pages 243–256, Albuquerque, 1996. Sandia National Laboratories.

[12] H.L. de Cougny and M.S. Shephard. Parallel refinement and coarsening of tetrahedral meshes. submitted, 1998.

[13] H.L. de Cougny and M.S. Shephard. Parallel unstructured grid generation. In J.F. Thompson, B.K. Soni, and N.P. Weatherill, editors, *CRC Handbook of Grid Generation*, pages 24.1–24.18, Boca Raton, 1999. CRC Press.

[14] H.L. de Cougny, M.S. Shephard, and C. Özturan. Parallel three-dimensional mesh generation on distributed memory (mimd) computers. *Engineering with Computers*, 12:94–106, 1996.

[15] K.D. Devine and J.E. Flaherty. Parallel adpaptive hp-refinement techniques for conservation laws. *Applied Numerical Mathematics*, 20:367–386, 1996.

[16] J.E. Flaherty, M. Dindar, R.M. Loy, M.S. Shephard, B.K. Szymanski, J.D. Teresco, and L.H. Ziantz. An adaptive and parallel framework for partial differential equations. In D. F. Griffiths, D. J. Higham, and G. A. Watson, editors, *Numerical Analysis 1997 (Proc. 17th Dundee Biennial Conf.)*, number 380 in Pitman Research Notes in Mathematics Series, pages 74–90, Harlow, 1998. Addison Wesley Longman.

[17] J.E. Flaherty, R. Loy, P.C. Scully, M.S. Shephard, B.K. Szymanski, J. Teresco, and L. Ziantz. Load balancing and communication optimization for parallel adaptive finite element methods. In *Proc. XVII International Conf. of the Chilean Comp. Sci. Soc.*, pages 246–255, Los Alamitos, 1997. IEEE Computer Society Press.

[18] J.E. Flaherty, R. Loy, M.S. Shephard, B.K. Szymanski, J. Teresco, and L. Ziantz. Adaptive local refinement with octree load-balancing for the parallel solution of three-dimensional conservation laws. *Journal of Parallel and Distributed Computing*, 47:139–152, 1997.

[19] J.E. Flaherty, R. Loy, M.S. Shephard, B.K. Szymanski, J. Teresco, and L. Ziantz. Predictive load balancing for parallel adaptive finite element computation. In H.R. Arabnia, editor, *Proc. Int. Conf. on Parallel and Distrib. Proc. Techs. and Applics. (PDPTA '97)*, pages 460–469, Las Vegas, 1997.

[20] J.E. Flaherty, R.M. Loy, C. Özturan, M.S. Shephard, B.K. Szymanski, J.D. Teresco, and L.H. Ziantz. Parallel structures and dynamic load balancing for adaptive finite element computation. *Applied Numerical Mathematics*, 26:241–265, 1998.

[21] J.E. Flaherty, R.M. Loy, M.S. Shephard, M.L. Simone, B.K. Szymanski, J.D. Teresco, and L.H. Ziantz. Distributed octree data structures and local refinement method for the parallel solution of three-dimensional conservation laws. In M.W. Bern, J.E. Flaherty, and M. Luskin, editors, *Grid Generation and Adaptive Algorithms*, volume 113 of *The IMA Volumes in Mathematics and its Applications*, pages 113–134, New York, 1999. Springer.

[22] J.E. Flaherty and P.K. Moore. Integrated space-time adaptive hp-refinement methods for parabolic systems. *Applied Numerical Mathematics*, 16:317–341, 1995.

[23] R. Garimella and M.S. Shephard. Tetrahedral mesh generation with multiple elements through the thickness. In *Proceedings 4th International Meshing Roundtable '95*, volume SAND 95-2130, pages 321–333, Albuquerque, 1995. Sandia National Laboratories.

[24] O. Klaas and M.S. Shephard. Automatic generation of partition of unity discretizations for three-dimensional domains. In *Modeling and Simulation-Based Engineering*, pages 89–94, Palmdale, 1998. Tech Science Press.

[25] O. Klaas and M.S. Shephard. Automatic generation of octee-based three-dimensional discretizations for partition of unity methods. Computer Methods in Applied Mechanics and Engineering, to appear, 1999.

[26] R.M. Loy. *Adaptive Local Refinement with Octree Load-Balancing for the Parallel Solution of Three-Dimensional Conservation Laws.* PhD thesis, Rensselaer Polytechnic Institute, Troy, 1998.

[27] C. Özturan. *Distributed Environment and Load Balancing for Adaptive Unstructured Meshes.* PhD thesis, Rensselaer Polytechnic Institute, Troy, 1995.

[28] M.S. Shephard, S. Dey, and J.E. Flaherty. Geometry-based issues associated with p-version finite element computations. *Computer Methods in Applied Mechanics and Engineering*, 150:39–50, 1997.

[29] M.S. Shephard, J.E. Flaherty, C.L. Bottasso, H.L. de Cougny, C. Özturan, and M.L. Simone. Parallel automated adaptive analysis. *Parallel Computing*, 23:1327–1347, 1997.

[30] M.S. Shephard, J.E. Flaherty, H.L. de Cougny, C.L. Bottasso C. Özturan, and M.W. Beall. Parallel automated adaptive procedures for unstructured meshes. Technical Report R-807, AGARD, Neuilly-sur-Seine, 1995. Special Course on Parallel Computing.

[31] M.S. Shephard, J.E. Flaherty, C.L. Bottasso H.L. de Cougny, and C. Özturan. Parallel automatic mesh generation and adaptive mesh control. In M. Papadrakakis, editor, *Solving Large Scale Problems in Mechanics: Parallel and Distributed Computer Applications*, pages 459–493, Chichester, 1997. John Wiley and Sons.

[32] M.S. Shephard and M.K. Georges. Reliability of automatic 3-d mesh generation. *Computer Methods in Applied Mechanics and Engineering*, 101:443–462, 1992.

[33] J.D. Teresco, M.W. Beall, J.E. Flaherty, and M.S. Shephard. A hierarchical partition model for adaptive finite element computation. *Computer Methods in Applied Mechanics and Engineering*, 1998. submitted.

[34] W.D. Turner, J.E. Flaherty, S. Dey, and M.S. Shephard. Multilevel preconditioned qmr methods for unstructured mesh computation. *Computer Methods in Applied Mechanics and Engineering*, 149:339–357, 1997.

4 Participating Scientific Personnel

The Principal Investigators are Joseph E. Flaherty and Mark S. Shephard. They were assisted by the following graduate scholars:

- Hugues L. de Cougny, Ph.D. Mechanical Engineering, *Parallel Unstructured Distributed Three-Dimensional Mesh Generation*, May 1998.
- Saikat Dey, Ph.D. Civil Engineering, *Geometry-Based Three-Dimensional hp-Finite Element Computation*, May 1997.
- Raymond M. Loy, Ph.D. Computer Science, *Adaptive Local Refinement with Octree Load-Balancing for the Parallel Solution of Three-Dimensional Conservation Laws*, May 1998.
- Can Özturan, Ph.D. Computer Science, *Distributed Environment and Load Balancing for Adaptive Unstructured Meshes*, May 1995.
- Wesely Turner, Ph.D. student, Computer Science.

5 Report of Inventions

None

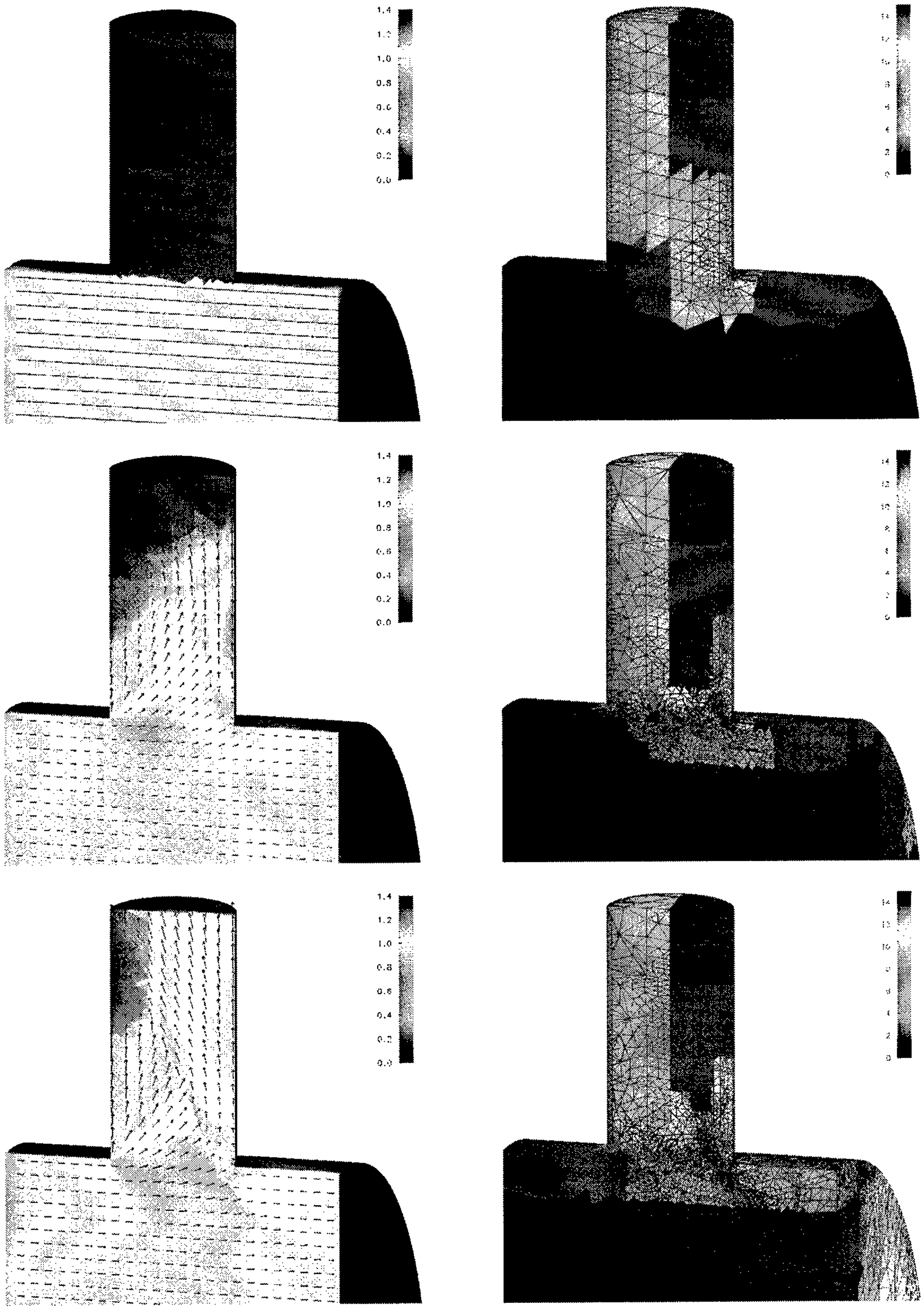

Figure 1: Projections of the Mach number and velocity vectors (left) and mesh and partitioning (right) onto the surfaces of a perforated cylinder at times 0, 0.3, and 0.6.